Part 1: Written Problems (50pts)

1: What is a default constructor? What is the consequence if a class does not have a default constructor?

A default constructor is a constructor that does not take any parameters and sets all the assets of the object to zero.

2: What is encapsulation? Why is it useful?
Encapsulation is the concepts of having data and functions that manipulate the data so that they cannot be affected by outside interference. This is useful because it allows for the creation of user created types also known as classes.

3: Write a function vector append(vector a, vector b) that appends one vector after another.

```
Vector append(vector a, vector b)
{

for(int i = 0; i < b.size();i++)
{

a.push_back(b[i]);

}

Return a;

}
```

Write a function vector merge(vector a, vector b) that merges two arrays, alternating elements from both arrays. If one array is shorter than the other, then alternate as long as you can and then append the remaining elements from the longer array. For example, if a is

```
Vector merge(vector a, vector b)
{

Vector c;

Int size = 0;
```

```
if(a.size() > b.size())
{
	Size = a.size();
}
Else
{
	Size = b.size();
}

for(int i = 0; i < size;i++)
{
if(i > b.size())
{
	for(int j = i; j < a.size();j++)
	{
		c.push_back(a[j]);
	}
}
else if(i > a.size())
{
	for(int t = i; t < b.size();t++)
	{
		c.push_back(b[t]);
	}
}
c.push_back(a[i]);
c.push_back(b[i]);
}

Return c;

}
```

System Manual:

Socialite Header File:

Public: constructors, mutators, inspectors.

Socialite(); :
This is used as the default constructor of the object.

Socialite(string lastName, string firstName, string userID, string picture, string website, string webdes); :
This can be used as an alternative constructor by including strings for the objects last name, first name, user ID, picture, website, and website description.

setLastName(string newName);
setFirstName(string newName);
setUserID(string newUserID);
setPicture(string newPicture);
setWebsite(string newWebsite);
setWebDes(string newDesc);
addClique(string newClique);

—
All of these are used to mutate an object's attributes, assigning the attribute named after "set" with the element passed to the method. The newest function "addClique" appends a clique onto the already existing vector of cliques.

string getLastName() const;
string getFirstName() const;
string getUserID() const;
string getPicture() const;
string getWebsite() const;
string getWebDes() const;
Int getNumCliques() const;
String getClique(int x) const;

—
These methods are used as inspectors to retrieve and show different attributes of the object it is used on. Returning the attribute that is named after “get” in the method name. The two newest inspectors “getNumCliques” and “getClique” are used to deal with the cliques that have been added to the socialites. getNumCliques returns the number of cliques that a socialite has, getClique returns the name of a clique that a socialite is in, specified by a number passed to the function that indicates the position of the clique name in the vector.

ostream & toString(ostream &out);
ostream & toHTML(ostream &out);

—
toString and toHTML are both meant to do the same thing but are implemented differently in the implementation file.

Private:

string lastName_;
string firstName_;
string userID_;
string picture_;
string website_;
string webdes_;
vector <string> cliques_;

Each of these strings are meant to hold information for the differently named attributes. With picture and website meant to hold links to an image and a website. The newest attribute “cliques” is meant to hold a vector of strings that are the names of cliques that socialite is in.

Socialite Implementation File:

Socialite::Socialite():

Used as the default constructor and automatically sets the attributes values to “” or null.

Socialite::Socialite(string lastName, string firstName, string userID, string picture, string website, string webdes):

Used as an alternative constructor and automatically sets the attributes values to the passed elements.

setLastName(string newName);
setFirstName(string newName);
setUserID(string newUserID);
setPicture(string newPicture);
setWebsite(string newWebsite);
setWebDes(string newDesc);
addClique(string newClique)

All of these are used to mutate an object’s attributes, assigning the attribute named after “set” with the element passed to the method all the same way. “addClique” appends a new string onto the vector of cliques on the socialite.

string getLastName() const;
string getFirstName() const;
string getUserID() const;
string getPicture() const;
string getWebsite() const;
string getWebDes() const;
getClique(int x) const;
getNumCliques() const;

These methods are used as inspectors to retrieve and show different attributes of the object it is used on. Returning the attribute that is named after “get” in the method name all the same way. “getNumClique” takes the size of the clique vector of a socialite and returns that. “getClique” uses a number passed to it to find a specific element in the clique vector and returns that string.

ostream & toString(ostream &out);
ostream & toHTML(ostream &out);

---

toString and toHTML both export an object but are formatted differently.

toString exports the object's attributes in a reader friendly way as if to be sent to a text file.

toHTML exports the object’s attributes so that it can be written to an HTML file and understood by a browser to display the information so that it is user friendly.

Main Implementation File:

Main();

The main program starts by declaring a stream to take in the socialite information. Then sends that file location to the readIn() function. The main then asks for a number to allow the user to manually end the program.

readIn(istream &in);

—
After first creating a temp string variable called line, the function then checks to see if the file passed to it is there by seeing if it has reached the end of the file by calling !in.eof() in a while loop. Then within the while loop it creates a socialite called “temp” to hold the information that is to be got from the file. readIn then gets the first 6 lines of code through a hard code of geting a line from the file and then setting that line as an attribute of the temporary socialite. After that a while loop takes care of the cliques that are to be read into the socialites until the break (“++++”) is found. The socialite is then sent to the screen using the “toString” function and then a file is created to send the socialite to an HTML file, as long as the file exists.